INTEPRINDEREA de CALCULATOARE ELECTRONICE

CERTIFICAT de GARANȚIE Nr.......... 36783

Produsul: EXTENSIE FLOPPY HC.

Seria: 39296

Data fabricației: 03 '94

Produsul este garantat pe o perioadă de 6 luni de la instalare și maxim 12 luni de la data vînzării. Acest termen se prelungește cu perioada de timp în care produsul a stat în reparație. Orice defecțiune care se datorează unei utilizări necorespunzătoare sau manipulari defectuoase sau neconformă cu instrucțiunile de instalare anulează garanția. Nu se consideră în garanție produsele care au fost desigilate de beneficiar. Cumpărătorul are obligația de a prezenta certificatul de garanție la întocmirea reclamației privitoare la defectarea produsului.

Produsele care sînt reparate în termenul de garanție de către întreprinderi neautorizate prin "agreement" de Intreprinderea de Calculatoare Electronice își pierd garanția.

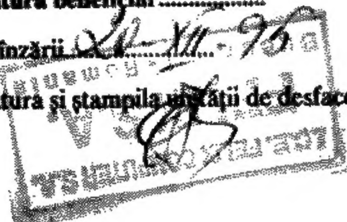
Șef Serviciu CTC

S-au făcut probe de funcționare, am primit manualul de funcționare și certificatul de garanție.

Semnătura beneficiar..................

Data vînzării 20-XII-95

Semnătura și stampila unității de desfacere



ice felix ®

COMPUTER S.A.

INTERFAȚA 1 HC

Manual de utilizare

CUPRINS

CAPITOLUL 1. INTRODUCERE

1.1. Prezentare generală

Interfaţa 1, pe scurt IF1, înglobează de fapt pe o singură plachetă de circuit imprimat trei interfeţe: cu discul flexibil (sau floppy), cu o linie serială standard CCITT V24 şi o interfaţă mai putin obişnuită care permite cuplarea mai multor HC-uri printr-o singura pereche de fire torsadate, conexiune denumită reţea.

Interfaţa de floppy permite cuplarea a pînă la doua minidrive-uri (5.25 ţoli) 40 de piste realizînd o capacitate formatată de 320 kilocteţi pe ambele feţe ale discului flexibil, spaţiu care poate fi utilizat pentru a memora pînă la 64 de fişiere distincte.

Rata medie de transfer a discului este de 25 pînă la 30 de ori mai mare decît a interfeţei standard de casetă magnetică. Dacă mai adaugăm la aceasta şi accesul aleator la informaţii (timpul maxim de acces la un sector de disc este de circa 1.7 secunde), este imposibil să nu remarcăm avantajele majore faţă de interfaţa de casetă.

Interfaţa serială rezolvă, în principal, problema cuplării unei imprimante la HC, dar poate fi folosită şi pentru a transfera datecu orice alt tip de calculator.

Interfaţa de reţea oferă o soluţie pentru una din aplicaţiile posibile ale calculatoarelor HC: învăţămîntul. Cuplate într-o reţea de pînă la 64 de sisteme, reţeaua HC-urilor dintr-o sală de laborator informatic poate uşura atît sarcina profesorului, cît şi sarcina elevilor.

Viteza de transfer a informaţiei prin reţea este de 80 kilocteţi pe secundă. Transferurile de date se fac în blocuri cu lungime variabilă (maxim 255 octeti), însoţite de blocuri de control care specifică adrese sursă/destinaţie, număr bloc, etc. Protocolul este suficient de cuprinzător pentru a permite schimbul simultan de mesaje între oricîte noduri ale reţelei folosind numai două fire torsadate pentru a lega nodurile între ele.

Din punct de vedere al programării, IF1 se integrează în sistemul BASIC al calculatorului HC, oferind fie noi instrucţiuni, fie extensii ale instrucţiunilor existente. Extinderea limbajului BASIC se face fără nici o modificare a plăcii de bază, interfaţa 1 interceptînd prin hardware rutina de eroare din placa de bază.

Cele trei interfeţe oferă pe lîngă facilităţi de încărcare salvare programe şi date, comenzi pentru manipularea de fişiere, ceea ce oferă o nouă dimensiune în stocarea şi regăsirea datelor folosind programe scrise în BASIC.

Interfaţa 1 se poate ataşa la orice calculator din seria HC (HC 85, HC 90, HC 91...) şi chiar la alte tipuri de calculatoare care au ca soft de bază Basic-Sinclair, cu amendamentul că magistrala de procesor să fie scoasă pe EDGE conector. Demn desubliniat este faptul că interfaţa poate fi acţionată şi în CP/M, dacă se cuplează la HC 91. Dacă aţi cuplat IF1 la un HC 91 şi doriţi să lucraţi în CP/M, consultaţi anexa 7 din prezentul manual.

1.2 Unitatea de disc flexibil

Interfaţa de disc flexibil este realizată cu un controler de disc flexibil 8272 (FDC) care asigură semnal de comandă pentru interfaţarea procesorului cu 2 unităţi de disc. Acest circuit poate lucra fie în format simplă densitate (FM IBM) sau în format dublă densitate (MFM), inclusiv dublă faţă.

Aşa cum am arătat în paragraful precedent, IF1 este dotată în general cu o unitate de disc flexibil de 320K.

Atentie! Nu porniţi sau opriţi calculatorul avînd discUL introdus în unitatea de disc. Informaţiile de pe disc pot fi distruse. Discul folosit este de 5'25" şi se pot folosi fie DD/DS (dublă densitate/dublă faţă) fie SD/SS (simplă densitate/simplă faţă).**Nu folosiţi discuri**

HD. Recomandăm dischete IZOT tip EC 5288 SS/SD, EC 5289 SS/DD, 5253 E etc.
Pentru a lucra cu discul trebuiesc știute următoarele lucruri:

I - Inserarea discului:
Discul se introduce în unitatea de disc cu eticheta în sus (cu gaura de index și cea de protecție la scriere în stînga). Dacă decupajul din stînga este acoperit, discul este protejat la scriere.

II - Manipularea discului:
a. Nu folosiți pix cu pastă sau creion pentru a scrie eticheta discului (folosiți creion tip carioca).
b. Nu puneți agrafe pe marginea discului.
c. Nu atingeți părți expuse ale suprafeței magnetice.
d. Nu încercați să curățați suprafața magnetică în niciunfel.
e. Nu apropiați magneți de disc.
f. Introduceți discul în plic dupăscoaterea din unitate.
g. Protejați discul de lichide , praf și scrum de țigară.
h. Păstrați discul la o temperatură între 10° C și 55° C și o umiditate relativă între 8% și 80%.

1.3. Punerea sub tensiune

Tensiunea de alimentare este de 220 Vef +10% - 15% cu o frecvență 50Hz +1Hz -1Hz. Puterea eléctrica absorbită de la rețea depinde de la un model la altul, fără a depăși 50W în configurație UC+IF1 cu o unitate de disc flexibil.
Modul de alimentare este diferit de la un model la altul.
Pentru exemplificare vă prezint cele două variante standard:

a: cu alimentare separată (HC85, HC90).

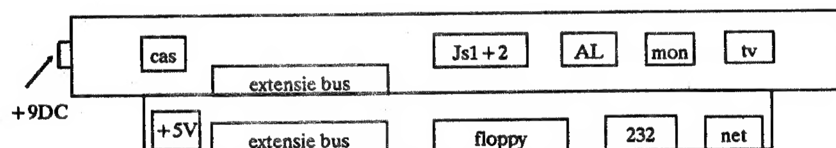


Fig. 1. Vedere din spate a calculatoarelor HC 85, HC 90.

Am folosit următoarele simboluri :
+9VDC - este o mufă jack mamă în care se introduce fișa alimentatorului de HC.
CAS - mufa pentru cuplat casetofonul.
Jsl + 2 - conector joystick (dublează tastele numerice)
MON - conector video: se pot cupla monitoare RGB alb negru sau color, monitor TTL sau video complex.
TV - mufa televizor canalul 8.
+5V - alimentator interfața 1 (de la UDF).
232 - interfața serială.
net - network rețea de 2-64 HC-uri.

Rețineți faptul că HC-ul nu funcționează dacă nu alimentați și interfața 1.

b: alimentare din aceeași sursă:

Calculatorul se alimentează cu + 9VDC (mufa AL) din UDF.
Această alimentare merge și la IF1 nemaifiind necesară alimentarea separat a IF1 cu +5V ca în fig. 1.
Este posibil ca o perioadă să vindem HC91 și Extensie cu alimentări separate.
În acest caz cei doi conectori vor fi diferențiați prin puncte de vopsea diferită: un punct pe carcasa conectorului și unul de aceeași culoare pe carcasa calculatorului.

CAPITOLUL 2. PRIMELE OPERAȚII CU MINIDISCUL

Auto-run

După ce ați terminat de instalat Interfața 1 și Minidrive-ul sînteți curios să aflați ce programe vă așteaptă pe discul de demonstrație. Pentru aceasta, inserați discheta în Minidrive (sau dacă aveți două Minidrive-uri, în Minidrive-ul 1), și introduceți:

NEW

urmat de:

RUN (și CR)

Aceste comenzi vor declanșa încărcarea automată și rularea primului program de pe floppy. După ce ați terminat de privit acest program, scrieți între ghilimelele din josul ecranului titlul unui program, apoi apăsați CR.

Catalogul

Pentru a afla ce alte programe se găsesc pe floppy-ul de demonstrație, introduceți instrucțiunea CATalog:

CAT 1

unde 1 identifică numărul Minidrive-ului pe care îl folosiți

În aproximativ 3 secunde pe ecranul televizorului se va afișa:
- un catalog al tuturor numelor fișierelor memorate pe dischetă
- spațiul rămas disponibil pe dischetă (în kiloocteți)

Încărcarea programelor

Următorul lucru de făcut este încărcarea programului pe care vreți să-l executați în continuare. Pentru asta alegeți mai întîi un program, apoi introduceți:

LOAD *"d";1;"nume"

unde "nume" este numele programului pe care l-aţi ales

"d";1 identifică ce Minidrive folosiţi

steluţa comunică calculatorului că folosiţi un Minidrive şi nu interfaţa obisnuită de casetă

După o scurtă pauză, ecranul va afişa mesajul OK (dar fără numele programului). Puteţi acum lansa programul în execuţie (cu RUN).

CAPITOLUL 3. UTILIZARE MINIDISC PENTRU PROGRAME

Salvare, verificare, încărcare şi comasare programe

În manualul de utilizare HC aţi găsit instrucţiunea SAVE, care salvează programe pe casetă. Salvarea programelor pe dischetă este la fel de simplă. Pentru exemplificare va fi folosit programul de mai jos, denumit Pătrate. El tipăreşte numerele de la 1 la 10 împreună cu pătratele lor.

```
10 REM Patrate
20 FOR n=1 TO 10
30 PRINT n,n*n
40 NEXT n
```

Pentru a salva acest program pe casetă, aţi fi introdus:

SAVE "Patrate"

Pentru a-l salva pe discheta din Minidrive-ul 1, introduceţi:

SAVE *"d";1;"Patrate"

După cîteva secunde în care marginea ecranului va clipi, programul va fi salvat. (Numele programelor memorate pe floppy pot avea o lungime maximă de 11 caractere).

Aşa cum probabil v-aţi imaginat deja, puteţi verifica corecta înregistrare a programului pe floppy introducînd:

VERIFY *"d";1;"Patrate"

Ecranul va afişa mesajul OK.

Puteţi încărca acum programul Patrate introducînd:

NEW

urmat de:

LOAD *"d";1;"Patrate"

În continuare, pentru a face ca programul să se lanseze automat, încercaţi să introduceţi:

SAVE *"d";1;"Patrate2" LINE 10

apoi:

NEW

şi apoi:

LOAD *"d";1;"Patrate2"

Minidrive-ul poate fi folosit şi pentru a comasa programele.

Introduceţi:

NEW

urmat de:

```
100 REM alte Patrate
110 FOR n=11 TO 20
120 PRINT n,n*n
130 NEXT n
```

şi acum introduceţi:

MERGE *"d";1;"Patrate"

şi programul Patrate va fi adăugat la listing.

Pe scurt, aşa cum v-aţi dat deja seama, sintaxa folosită pentru obişnuita interfaţă de casetă (explicată în secţiunea "Memorare pe bandă" din manualul de programare BASIC) se aplică şi la Minidisc.

Ştergerea programelor

Să presupunem că aţi terminat de lucrat cu programul Patrate. Pentru a-l şterge, introduceţi:

ERASE "d";1;"Patrate"

(Ca înainte, "d";1 indică ce Minidrive folosiţi).

În timpul execuţiei intrucţiunii ERASE, marginea ecranului va clipi.

Formarea discurilor

Înainte de prima utilizare a unei dischete, inseraţi-o într-un Minidrive (de exemplu Minidrive-ul 1) şi introduceţi:

FORMAT "d";1

"d";1 identifică Minidrive-ul pe care îl folosiţi (în acest caz Minidrive-ul 1).

Formarea unui disc durează aproximativ treizeci de secunde. În timpul acesta, marginea ecranului se va schimba la început, şi va reveni puţin înainte de afişarea mesajului OK. Procesul de formare constă din iniţializarea fiecărei piste de pe dischetă, prin scrierea cîmpurilor de identificare şi date corespunzătoare fiecărui sector. După formarea unei piste, fiecare sector în parte este citit, verificînd suma de control. Mesajul OK apare numai dacă toate sectoarele au putut fi citite corect (nu se acceptă discuri cu sectoare eronate).

Formarea unei dischete nu trebuie repetată niciodată, şi pentru ca prin formarea unui disc se pierde orice a fost înregistrat pe el.

Apăsaţi acum:

CAT 1

unde 1 identifică numărul minidrive-ului pe care îl folosiţi

După cîteva secunde, în care timp marginea ecranului va clipi, va apare mesajul de eroare:

File not found

care semnifică faptul că floppy-ul nu conţine nici un program.

Capacitatea unui disc este de 318 kilocteţi, plus 2 kilocteţi pentru zona de catalog, gestionată de sistem.

Instalarea facilităţii de auto-run

Puţin mai înainte aţi folosit facilitatea de auto-run pentru discul de demonstraţie. Dacă aveţi un program pe care îl folosiţi adesea, vă puteţi stabili propria facilitate de auto-run, astfel încît să nu mai introduceţi instrucţiunile LOAD şi RUN. Acestea sînt regulile de urmat:

- programul trebuie să aibă numele run;
- discheta trebuie folosită în Minidrive-ul 1;
- facilitatea trebuie folosită fie imediat după punerea sub tensiune, sau imediat după comanda NEW.

Astfel, introduceţi programul respectiv, urmat de comanda:

SAVE *"d";1;"run" LINE număr

unde număr este numărul liniei de start. Numele run trebuie introdus literă cu literă. Nu apăsaţi tasta RUN.

Acum introduceţi:

NEW

urmat de:

RUN

adică tasta RUN, şi nu numele programului.

CAPITOLUL 4. DATE, CANALE ŞI CĂI

Precum ştiţi, un program este un set de operaţii care se execută atunci cînd apăsaţi RUN. Datele, pe de altă parte, sînt orice colecţie de litere, numere sau simboluri cu care poate lucra un program. Exemple sînt numerele de la 1 la 10 şi pătratele lor.

Datele pot fi trimise, sau recepţionate, către de la diferite părţi ale unui sistem de calcul. Aceste părţi sînt denumite 'canale'. Canalele către care se pot trimite date sînt:

- ecranul televizorului
- un fisier pe dischetă
- un alt calculator HC 91, dacă amîndouă calculatoarele sînt cuplate printr-o reţea.
- interfaţa RS232 şi de acolo, de exemplu, la un modem sau o imprimantă.

Canalele de la care se pot primi date sînt:

- claviatura
- un fisier pe dischetă
- un alt calculator HC 91, dacă amîndouă calculatoarele sînt cuplate printr-o reţea.
- interfaţa RS232, adică un modem sau un terminal.

Nodurile de comunicaţie dintre programul BASIC şi canale sînt denumite căi. În sistemul HC, numărul acestor căi este fixat la 16. Ele sînt numerotate de la 0 la 15, iar numerele de cale sînt întotdeauna precedate de semnul #.

Patru dintre aceste căi sînt deja cuplate la următoarele canale:

calea #0	trimite date către partea de jos a ecranului TV şi
calea #1	primeşte date de la claviatură;
calea #2	trimite date către partea de sus a ecranului TV, dar nu poate primi date;
calea #3	trimite date către imprimantă, dar nu poate primi date.

Orice intrucţiune care execută un transfer de intrare/ieşire foloseşte una din aceste căi în mod implicit. De exemplu, instrucţiunea PRINT foloseşte calea #2, iar instrucţiunea LPRINT foloseşte calea #3. Astfel, dacă introduceţi:

PRINT "Acesta este un calculator HC"

folosiţi de fapt o prescurtare a instrucţiunii:

PRINT #2;"Acesta este un calculator HC"

Verificaţi prin introducerea celor două forme.

Puteţi, totuşi, să faceţi fiecare instrucţiune să folosească o altă cale prin introducerea semnului # urmat de un număr de cale. Încercaţi să introduceţi:

LPRINT #2;"Acesta este un calculator HC"

în loc să fie trimis la imprimantă, acest mesaj apare pe ecranul TV.

Dar în loc să folosiţi căile prestabilite, puteti crea unele proprii. Căile #4 pînă la #15 sînt rezervate pentru acest scop; şi exista diferite 'specificatoare de canale' care indică perifericul dorit. Cîteva exemple sînt:

"K" pentru claviatura
"S" pentru ecran
"P" pentru imprimanta

(altele vor fi introduse mai tîrziu).

Remarcaţi faptul că K, S şi P sînt toate canale prestabilite. Ele solicită utilizarea virgulelor (,) drept separatori în instrucţiunile OPEN #. Dar cu alte canale puteţi folosi fie virgule fie punct-virgulă (;).

Pentru a creea o cale proprie folosiţi instructiunea OPEN #. De exemplu introduceţi:

10 OPEN #4,"S"

Astfel deschideţi calea #4 şi îl cuplaţi la canalul "S". Acum introduceţi:

20 PRINT #4;"Acesta este un calculator HC"

şi din nou linia va apare pe ecran. (Nu se recomandă deschiderea căilor 0, 1 sau 2, pentru că rezultatele acestor operaţii pot fi imprevizibile).

CAPITOLUL 5. FIŞIERE DE DATE PE DISC

Deschiderea unui fişier de date

Memorarea informaţiilor pe dischetă se face în fişiere. Fiecare fişier primeşte la creare un nume, pentru a putea fi regăsit mai tîrziu. Instrucţiunea care deschide şi denumeşte un fişier de date are întotdeauna aceeaşi forma. De exemplu testaţi instrucţiunea:

OPEN #4;"d";1;"Numere"

unde "Numere" este numele fişierului. Acesta poate fi orice şir de caractere de lungime maxim 11.

"d";1 identifică Minidiscul pe care îl folosiţi
numărul de cale (#) poate fi orice număr între 0 şi 15

Această instrucţiune face două lucruri distincte:

- stabileşte un canal cuplat cu fişierul: "d";1;"Numere"
- ataşează acest nou canal la calea #4.

Operaţia va dura cîteva secunde, în care timp calculatorul va căuta pe dischetă un fişier

cu numele "Numere". Pentru că nu exista fisierul "Numere", deschide canalul pentru scriere. (Daca ar fi găsit un fişier cu numele "Numere", l-ar fi deschis pentru citire).

Introducerea datelor

Odată ce aţi deschis un fişier, puteţi introduce date. Să presupunem că vreţi să memoraţi numerele de la 1 la 10 împreună cu pătratele lor. Introduceţi şi rulaţi programul următor:

```
10 FOR n=1 TO 10
20 PRINT #4;n'n*n
30 NEXT n
```

S-ar putea să credeţi că toate numerele au fost deja memorate pe dischetă. Dar de fapt calculatorul nu transfera în mod automat datele pe dischetă decît după ce s-a acumulat o anumită cantitate de informaţii, pe care o transferă dintr-odată. Acest procedeu se numeşte 'blocarea' datelor. Un bloc de date pe dischetă are lungimea de 256 de octeţi (sau caractere).

Pentru a memora pe floppy datele introduse trebuie să închideţi fişierul. Pînă nu faceţi acest lucru, nu veţi putea să citiţi din fişier.

Închiderea unui fişier

Închiderea unui fişier asigură memorarea definitivă a datelor pe dischetă. Închide de asemenea canalul (în cazul nostru "d";1;"Numere") şi detaşeaza calea (în cazul nostru #4) de la orice canal. Pentru a închide un fişier trebuie doar să închideţi calea asociată:

CLOSE #4

Marginea ecranului va clipi pentru a arăta că se înregistrează ceva pe dischetă. (Remarcaţi faptul că, la fel ca la instrucţiunea OPEN, instrucţiunea CLOSE este urmată în mod automat de #).

Căile #0, #1, #2, #3 rămîn întotdeauna ataşate unui canal, chiar daca se executa o instructiune CLOSE specifică. Dacă încercaţi să închideţi una din aceste căi, căile #0 şi #1 se vor ataşa automat la canalul K; calea #2 la canalul S; iar calea #3 la canalul P.

Citirea datelor dintr-un fişier

Pentru a citi datele din fişierul "Numere" rulaţi următorul program:

```
10 OPEN #4;"d";1;"Numere"
20 FOR b=1 TO 10
30 INPUT #4;m;n
40 PRINT "Patratul lui ";m;" este ";n
50 NEXT b                 |    |    |
60 CLOSE #4               +--- +--- +--- lăsaţi spaţii
RUN
```

Pentru că fişierul "Numere" există deja pe dischetă, canalul "d";1;"Numere" este deschis pentru intrare, şi orice încercare de a scrie date ar fi generat o eroare.

Se poate de asemenea folosi funcţia INKEY$ pentru a citi date dintr-un fişier (întoarce întotdeauna următorul caracter din fişier). Încercaţi programul următor:

```
10 OPEN #11;"d";1;"listing"
20 LIST #11
30 CLOSE #11
40 OPEN #12;"d";1;"listing"
50 PRINT INKEY$#12;
60 GO TO 50
```

Acest program se va termina cu un mesaj de sfîrşit de fişier, adică End of file.

Observaţii asupra lui PRINT şi INPUT

Pentru că instrucţiunile PRINT şi INPUT au fost concepute în principal pentru utilizarea cu ecranul şi claviatura, trebuie să fiţi atenţi la folosirea lor cu fişiere.

'separatori'

Instrucţiunea PRINT are trei forme de separatori:

- semnul ; (punct-virgulă) nu tipăreşte nimic,
- semnul , (virgulă) va aduce la începutul următoarei jumătăţi de linie,
- semnul ' (apostrof) sare la linie nouă (codul CR).

Instrucţiunea INPUT aşteaptă întotdeauna să introduceţi CR după un număr sau un şir. Astfel, de fiecare dată cînd tipăriţi într-un fişier din care vreţi să citiţi mai tîrziu cu INPUT, trebuie fie să:

- tipăriţi fiecare element separat, adică

```
10 PRINT #4;2
20 PRINT #4;3
```

sau

- separaţi elementele cu apostrof, adică

```
10 PRINT #4;2'3
```

De asemenea, în instrucţiunile INPUT, trebuie să folosiţi cu atenţie separatorii. Aşa cum ştiţi, INPUT poate tipări în partea de jos a ecranului orice se poate pune într-o instrucţiune PRINT. Dar dacă citiţi cu INPUT dintr-un fişier, fişierul se deschide numai pentru citire. Aşa încît, dacă includeţi orice s-ar fi tipărit la utilizarea ecranului, veţi obţine mesajul de eroare **Writing to a 'read' file** (Scriere într-un fişier de citire). Aceasta înseamnă că elementele dintr-o instrucţiune INPUT trebuie separate numai prin punct-virgulă, adică

```
10 INPUT #4;a;b
```


Atenţie de asemenea la citirea cu INPUT a şirurilor de caractere care conţin " (ghilimele), pentru că INPUT va interpreta ghilimelele drept sfîrşit de şir. Metoda de a evita acest lucru este de a înlocui, de exemplu:

```
10 INPUT #4;a$
```

cu

```
10 INPUT #4; LINE a$
```

Schimbarea căilor

Instrucţiunile PRINT pot conţine informaţii pentru mai multe căi la un moment dat. Programul următor va tipări "unu" pe ecran; "doi" într-un fişier pe dischetă denumit "cifre"; "trei" către staţia 1 pe reţea (vezi capitolul următor); şi "patru" în următoarea linie din ecran.

```
10 OPEN #4;"d";1;"cifre"
20 OPEN #5;"n";1
30 PRINT "unu";#4;"doi";#5;"trei"'#2;"patru"
40 CLOSE #4
50 CLOSE #5
```

'schimbarea culorilor'

După ce aţi folosit un canal diferit de ecran, se poate ca instrucţiunile PAPER şi INK să nu aibă nici un efect. Pentru a evita acest lucru, introduceţi:

```
PRINT;
```

înainte de a schimba PAPER sau INK.

Afişarea catalogului de fişiere

Pe măsura ce se înregistrează fişiere pe dischetă, acestea sînt introduse în mod automat în catalog. Astfel, pentru a afla ce fişiere sînt înregistrate pe dischetă, este suficient să inseraţi discheta într-un Minidrive şi să introduceţi instrucţiunea CATalog. De exemplu, introduceţi:

```
CAT 1
```

Ecranul televizorului va afişa:

- numele fişierelor
- spaţiul disponibil rămas pe dischetă (în kiloocteţi)

Puteţi să transferaţi ieşirea unui CAT către o cale introducînd:

```
CAT #număr;număr
```

unde primul număr este cel de cale, iar al doilea cel de Minidrive.

Aceasta vă permite să trimiteţi catalogul către o imprimantă, sau către un fişier, astfel încît să poată fi folosit de un program.

Protejarea unui fişier

Dacă doriţi ca un nume să nu apară în catalog, îl puteţi proteja dîndu-i un nume care are în poziţia 10 codul caracterului dorit plus 128. Introduceţi acest program:

```
10 OPEN #4,"d";1;"Rezultate" + CHR$ (128 + CODE " ")
20 FOR n=1 TO 15
30 PRINT #4;n'n*n
40 NEXT n
50 CLOSE #4
```

Acum introduceţi:

```
CAT 1
```

Numele fişierului nu va apare. Astfel că, de fiecare dată cînd generaţi un fişier protejat, amintiti-va sa-i notati numele undeva, pentru cazul în care îi uitaţi numele!

Extinderea unui fişier

Să presupunem că vreţi să extindeţi fişierul "Numere" pentru a include pătratele numerelor de la 1 la 20 în loc de numai 1 la 10. Un fişier nu poate fi redeschis pentru scriere, astfel că trebuie să:

- **creaţi o nouă versiune cu alt nume;**
- **transferaţi vechiul fişier în noua versiune;**
- **adăugaţi noile date**
- **închideţi vechiul fişier.**

Iată cum se poate face aceasta.
Mai întîi rulaţi acest program:

```
10 OPEN #4;"d";1;"Numere": REM pentru citire
20 OPEN #5;"d";1;"Numere 1": REM pentru scriere
30 FOR f=1 TO 10
40 INPUT #4;m;n
50 PRINT #5;m'n
60 NEXT f
70 FOR n=11 TO 20
80 PRINT #5;n'n*n
90 NEXT n
100 CLOSE #4; CLOSE #5
```

Pentru a verifica existenţa a două fişiere, "Numere" şi "Numere 1", introduceţi:


```
CAT 1
```

Apoi, ca să ştergeţi vechiul fişier, introduceţi:

```
ERASE "d";1;"Numere"
```

Pentru a verifica ştergerea, introduceţi:

```
CAT 1
```

Numele fişierului "Numere" a dispărut din catalog, iar noul fişier, "Numere 1" conţine acum numerele de la 1 la 20.

CAPITOLUL 6. REŢEAUA LOCALĂ

Configurarea unei reţele

Reţeaua locală permite utilizatorului şi prietenilor lui să schimbe între ei programe şi date. Aceasta înseamnă că numai unul dintre voi trebuie să introducă un program. O reţea este foarte utilă şi dacă numai unul dintre voi are un Minidisc.

Folosind cablul furnizat odată cu interfaţa, puteţi lega de la două pînă la 64 de calculatoare HC 91.

Configuraţia reţelei nu trebuie să fie în nici un caz o buclă închisă: calculatoarele de la capetele reţelei nu trebuie să fie conectate între ele. Fiecare capăt de reţea trebuie să aibă un conector neocupat.

NU PORNIŢI ŞI NU OPRIŢI NICIODATĂ UN HC CARE ESTE CUPLAT LA REŢEA ÎN TIMP CE ARE LOC UN TRANSFER DE DATE PE REŢEA.

Totuşi puteţi avea un HC oprit pe reţea; puteţi de asemenea să porniţi sau să opriţi HC-uri care sînt pe reţea, cu condiţia ca să nu se facă transferuri pe reţea în acel moment.

După ce aţi stabilit o reţea, fiecare calculator (sau staţie) trebuie să primească un număr de identificare diferit. Mai întîi stabiliţi împreună cu prietenii dumneavoastră, care va fi numărul fiecărei staţii, după care fiecare dintre dumneavoastră trebuie să introducă:

```
FORMAT "n";număr
```

unde "n" este numărul de staţie pe care l-aţi ales.

Dacă reţeaua este formată numai din două calculatoare, amîndouă pot folosi acelaşi număr de staţie. Şi pentru că amîndouă calculatoarele devin staţia 1 în mod automat la punerea sub tensiune, utilizarea instrucţiunii FORMAT nu mai este necesară.

Programele şi reţeaua

Să presupunem că aţi cuplat două calculatoare într-o reţea, cu numerele de staţie 1 respectiv 2.
Să presupunem că vreţi să trimiteţi către staţia 2 următorul program:

```
10 REM patrate
20 FOR n=1 TO 10
30 PRINT n,SQR n
40 NEXT n
```

Introduceţi programul urmat de:

```
SAVE *"n";2
```

(Remarcaţi că reţeaua nu foloseşte nume pentru programe.)
Între timp la staţia 2 trebuie introdus:

```
FORMAT "n";2
```

urmat de:

```
LOAD *"n";1
```

Staţia 2 va avea acum o copie a programului. Remarcaţi cum marginea ecranului rămîne neagră în timp ce calculatorul aşteaptă să salveze sau să încarce programul prin reţea. Staţia 1 nu va trimite pînă cînd staţia 2 nu este gata, iar staţia 2 va aştepta pînă cînd se emite ceva. Încercaţi să introduceţi linia cu SAVE înainte ca la staţia 2 să introduceţi LOAD şi vice versa.

Pentru a verifica transmisia corectă a programului, la staţia 2 trebuie introdus:

```
VERIFY *"n";1
```

în timp ce la staţia 1 se repetă transmisia programului introducînd:

```
SAVE *"n";2
```

SAVE este de fapt singura instrucţiune care transmite programe în reţea. Instrucţiunile LOAD, VERIFY şi MERGE sînt toate metode de a recepţiona programele.
Jocul de reţea din Anexa 1 este un bun exemplu pentru utilizarea programelor în reţea.

Fişiere de date în reţea

Să presupunem că doriţi să transmiteţi acum date către staţia 2. Instrucţiunea OPEN #4;"n";2 deschide un canal către staţia 2 pe reţea şi ataşează calea #4 la el, astfel încît dacă scrieţi prin calea #4, mesajul va fi pus pe reţea împreună cu o notiţă care indică sursa mesajului.
Dacă aţi fi introdus INPUT #4;m$ calculatorul dumneavoastră ar fi aşteptat informaţii adresate staţiei 1 de la staţia 2.

Acum introduceti acest program:

```
10 OPEN #4;"n";2: REM pentru iesire
20 INPUT a$: PRINT #4;a$
70 GO TO 10
```

Apoi introduceţi:

```
SAVE *"n";2
```

Acum introduceţi la staţia 2:

```
FORMAT "n";2
LOAD *"n";1
```

Introduceţi la staţia 1 RUN, iar la staţia 2 editaţi liniile 10 şi 40 pentru a se referi la staţia 1 şi nu la staţia 2. Apoi se introduce la staţia 2:

```
GO TO 40
```

Sînteţi acum gata să începeţi o conversatiela staţia 1 şi nu la staţia 2.
Apoi se introduce la staţia 2:

```
GO TO 40
```

Sînteţi acum gata să începeţi o conversaţie. Dar înainte de a face asta ar trebui să ştiţi cîteva lucruri.
-Tot ce tipăriţi prin calea #4 este blocat: adică nu este imediat pus pe reţea, ci se aşteaptă pînă cînd se acumulează o anumită cantitate de date. Aşa că este necesară închiderea canalului prin CLOSE, imediat ce aţi terminat de tipărit. Astfel se transmite zona tampon chiar dacă nu este plină. (Zona tampon are lungimea 255 de octeţi sau caractere).
- Tot ce tipăriţi este marcat ca sosind în mod specific de la staţia la care lucraţi, astfel încît dacă staţia 2 este în aşteptare pentru un mesaj de la altă staţie, mesajul dumneavoastră va fi ignorat. Dacă mesajul emis este ignorat, ecranul nu va afişa mesajul OK, şi marginea ecranului va rămîne neagră pîna cînd mesajul este emis şi se primeşte confirmare pozitivă de la staţia 2.
- În timp ce instrucţiunea INPUT poate fi folosită pentru a aştepta ca ceva să fie transmis, funcţia INKEY$ poate fi folosită pentru a citi reţeaua. Se va întoarce cu primul octet din orice a fost transmis sau din orice aşteapta să fie transmis. Altfel se întoarce cu şirul vid. Aceasta se numeşte interogare (polling).

Programul de mai jos va tipări orice este transmis către staţia 1:

```
10 OPEN #8;"n";1
20 PRINT INKEY$#8;
30 GO TO 20
```

(Pentru mai multe detalii despre INKEY$ vezi secţiunea 3.15 din manualul tehnic HC.)

Emisie generală

Există un număr de stație special, al cărui specificator este "n";0. Atunci cînd se așteaptă date de la stația 0, veți recepționa orice mesaj care este emis către stația 0. Iar atunci cînd scrieți, mesajul emis către stația 0 va fi recepționat de oricine citește date de la un canal cu specificatorul "n";0.

Aceasta ar fi foarte util, de exemplu, într-o școală dacă fiecare elev ar avea un calculator, dar numai profesorul ar avea un minidrive.

Să presupunem că profesorul dorește să emită un program. Mai întîi toți elevii ar trebui să introducă:

```
LOAD *"n";0
```

Această comandă va face ca toate calculatoarele elevilor să intre în așteptare pentru recepția programului. Profesorul ar trebui să salveze programul în rețea introducînd:

```
SAVE *"n";0
```

Emisiile generale, spre deosebire de mesajele private, încep imediat fără să aștepte ca alte calculatoare să fie gata să le recepționeze.

De asemenea, la emisia generală, calculatorul nu vă poate informa dacă mesajul emis a fost recepționat de către cineva.

Funcția INKEY$ nu poate fi folosită pentru a interoga un canal de recepție generală. La fel ca INPUT, ea va aștepta pur și simplu ca ceva să fie emis.

CAPITOLUL 7. UTILIZAREA INTERFEȚEI SERIALE

Conectarea perifericelor la interfața serială

Precum știți, setul de caractere al HC-ului conține atît simboluri simple (litere, cifre, etc.) cît și cuvinte cheie (instrucțiuni, nume de funcții, etc.). Toate aceste caractere pot fi emise și recepționate prin interfața serială către/de la orice dispozitiv compatibil; de exemplu o imprimantă, un modem sau o alta interfață serială conectată la un tip diferit de calculator.

Pentru a conecta oricare din aceste periferice la interfața serială, trebuie să folosiți un cablu cu un conector cu 9 pini la capătul dinspre interfața 1 și un conector corespunzător dispozitivului la care vă cuplați la celălalt capăt. (Pentru detalii de interconectare vezi Anexa 4.)

Apoi, înainte de a folosi interfața serială, va trebui să stabiliti modul de lucru al perifericului:

- modul 'auto line feed' trebuie dezactivat. (HC va emite secvența 'retur car' (CR) și 'avans rînd' (LF) pe un canal "t", dar numai 'retur car' (CR) pe un canal "b". Aceste canale "t" și "b" sînt explicate mai jos.)
- paritatea trebuie dezactivată.
- numărul de biți trebuie stabilit la 8 (opt).
- numărul de biți de stop trebuie stabilit la 1 (unu).
- viteza de emisie/receptie (adică numărul de biți pe secundă). HC-ul poate comunica la oricare din vitezele standard, adică:

50, 110, 300, 600, 1200, 2400, 4800, 9600, 19200.

Este bine să folosiți cea mai mare viteză pe care o permite perifericul la care vă cuplați. (Veți vedea mai jos cum puteți face ca HC-ul să folosească aceeași viteză.)

În astfel de momente, un manual de instalare pentru perifericul la care vă cuplați este foarte util.

Canalele t și b

Interfața serială are două tipuri de canale: canalul t și canalul b.

Canalul t

Canalul t (de la text) este folosit de obicei pentru a trimite listinguri. Canalul t are următorul efect asupra setului de caractere:

cod caracter	
0 - 31	(caracterele de control) nu sînt emise, cu excepția lui 13 (retur car) care este trimis ca 13 urmat de 10 (avans rînd)
32 - 126	(caracterele tipăribile) sînt trimise ca atare
127 - 164	(caracterele grafice) nu sînt emise. Ele sînt înlocuite de caracterul ? (codul 63)

Pentru INPUT și INKEY$ canalul t transferă numai caractere pe 7 biți, așa încît forțează la 0 bitul 7.

Pentru a folosi canalul t, trebuie mai întîi să stabiliți viteza de lucru pe sîrme. Așa că introduceți:

```
10 FORMAT "t";viteza
```

unde viteza este viteza de lucru pe care ați stabilit-o și la periferic.

Acum, pentru a deschide o cale către canalul t, introduceți:

```
20 OPEN #3,"t"
30 LLIST
```

Marginea ecranului va clipi și listingul va fi trimis către periferic. (Remarcați că LLIST este este o prescurtare de la LIST #3.) Introduceți acum:

```
LPRINT "Acesta este un mesaj."
```

Și acest mesaj va fi trimis către dispozitiv.

Dacă HC este cuplat cu un terminal sau un calculator care poate trimite caractere, atunci puteți citi date de la terminal sau calculator. Introduceți:

```
10 FORMAT "t";viteza
20 OPEN #4,"t"
30 PRINT INKEY$#4;
40 GO TO 30
```

Acum, orice caracter primit de la terminal sau calculator va fi afișat pe ecran.

Canalul b

Canalul b (de la binar) trimite toţi cei 8 biţi ai codurilor folosite de HC, şi vă permite să trimiteţi coduri de control către imprimante etc.
Şi la INPUT şi INKEY$ canalul b întoarce caractere pe 8 biţi.
SAVE şi LOAD funcţionează numai cu canalul b.
Dacă aţi conectat două HC-uri prin interfaţa serială sau doriţi să vă memoraţi programele pe un alt tip de calculator care are de asemenea o interfaţă serială, veţi dori să salvaţi şi să încărcaţi programe prin interfaţa serială. Pentru aceasta introduceţi:

```
FORMAT "b",viteza
```

Acum puteţi încerca, de exemplu:

```
10 REM cifre
20 FOR n=1 TO 10
30 PRINT n,n*RND
40 NEXT n
```

urmat de:

```
SAVE *"b"
```

La celălalt capăt al legăturii cineva trebuie să introducă:

```
LOAD *"b"
```

Extensiile uzuale sînt de asemenea posibile:

```
SAVE *"b";SCREEN$
```

şi:

```
SAVE *"b";LINE numar
```

Cum se trimit coduri de control

Multe imprimante primesc secvenţe de control pentru operaţii de genul tipărire cu lăţime dublă. Pentru a trimite caracterele de control trebuie să folosiţi canalul b. Atenţie însă, prin canalul b returul de car (CR) nu este urmat automat de avans rînd (LF). De aceea este preferabil să aveţi două canale deschise, unul b şi altul t: veţi folosi canalul b pentru a trimite secvenţele de control şi canalul t pentru texte. Să presupunem că secvenţa de control pentru imprimare cu lăţime dublă este 14. Introduceţi:

```
10 OPEN #4;"b"
20 PRINT #4;"Latime normala ";
30 PRINT #4;CHR$ 14;"Latime dubla"
40 CLOSE #4
```

(Dacă exemplul nu funcţionează, căutaţi în manualul imprimantei codul pentru lăţime

dublă.)

Încercaţi şi exemplul de mai jos:

```
10 OPEN #5;"b"
20 OPEN #6;"t"
30 PRINT #5; CHR$ 14;
40 LIST #6
50 CLOSE #5: CLOSE #6
```

Acest exemplu ar trebui să producă un listing pe lăţime dublă.

Caracteristici apărute la REV.2 la Interfaţa 1

Revizia 2 de la interfaţa 1 constă doar din modificări ale softului din interfaţa RS 232. Modificările sînt în principal următoarele:

- Îmbunătăţirea substanţială a rutinei de tipărire pe imprimantă, adică a rutinei de ieşire pe canalul 't'.
- Eliminarea unor erori găsite în softul Interfeţei 1 REV.1
- Adăugarea unui hook code.

Modificări rutina imprimantă

Controlul lăţime imprimare. În mod implicit înainte de tipărirea coloanei 81 a unei linii se emite secvenţa (CR)(LF) care forţează o linie nouă la imprimantă şi se resetează controlul intern de coloane. În acest fel lăţimea LIST-ingului program este limitată la 80 de coloane. De asemenea PRINT-urile foarte lungi sînt continuate pe linia următoare. Octetul 23729 din variabilele extinse contine lăţimea imprimantei în coloane (implicit 80). Inainte de a modifica prin POKE această valoare trebuie să vă asiguraţi că variabilele extinse sînt inserate în sistem şi că aveţi REV.2 a Interfeţei 1. După instrucţiunile:

```
CLOSE #0: LET V=PEEK 23729
```

variabila v va conţine:

- 0 pentru REV.1
- 80 pentru REV.2

După verificarea versiunii puteţi de exemplu modifica lăţimea imprimantei cu instrucţiunea: POKE 23729,64 la 64 de coloane.

Extinderea setului de caractere recunoscute cu operatorii TAB şi AT. Operatorul asigură tabularea datelor din 8 în 8 coloane.
Operatorul TAB exp aduce poziţia de imprimare în coloana exp.
Operatorul AT lin,col aduce poziţia de imprimare în coloana col. lin este ignorat. Numerotarea coloanelor pe imprimantă începe cu 0. Operatorii INK, PAPER, INVERSE, FLASH sînt ignoraţi.

Corecții erori

La închiderea unui stream de tip 't' se emite (CR)(LF) iar la închiderea unui stream de tip 'b' nu se emite nimic. Interfața veche emite un (CR) indiferent de tipul canalului 'b' sau 't'.

La revizia 1 dacă operația de închidere a unui stream era întreruptă cu BREAK de la tastatură, în memorie rămîne o zonă cu octeți nefolositori. La rev. 2 pana a fost eliminată.

Hook code nou

A fost inserată o noua funcție hook code: creează canal ad-hoc.

- Hook code +3C
- Intrarea 23769 tip canal: b/t
- Iesirea reg de adresă canal.

Canalul fiind ad-hoc el este distrus în mod automat la revenirea în mod comandă.

CAPITOLUL 8. INSTRUCȚIUNEA MOVE

Pînă acum am făcut transferuri de date de la un program către un canal sau invers. Instrucțiunea MOVE vă permite să mutați date de la un canal la altul. De exemplu, pentru a muta date de la claviatură la ecran, introduceți:

```
10 MOVE #1 TO #2
```

apoi:

```
RUN
```

Orice veți introduce de la claviatura va apare pe ecran. Dar veți descoperi că apăsarea lui BREAK nu face decît să tipărească un spațiu pe ecran. Pentru a ieși din această capcană, apăsați CR pînă ce ajungeți la ultima linie din ecran. Apoi, răspundeți cu BREAK la întrebarea scroll? (Ar trebui ca pe viitor să evitați să mutați date de la claviatură la orice altă cale pentru că s-ar putea să nu mai reușiți să ieșiți din instructiunea MOVE.)

Instrucțiunea MOVE se mai poate utiliza și pentru a examina fișierele memorate pe floppy. De exemplu, dacă mai aveți pe floppy fișierul "Numere" (vezi pagina 7), îi puteți examina conținutul cu instrucțiunea:

```
10 MOVE "d";1;"Numere" TO #2
```

(Remarcați că nu trebuie să deschideți/închideți (OPEN/CLOSE) fișierul, MOVE face singur lucrul acesta.)

De asemenea, pentru a face o copie a programului "Numere" introduceți:

```
10 MOVE "d";1;"Numere" TO "d";1;"Numere 2"
```

În acest caz, MOVE deschide o cale pentru a citi din fișierul existent ("Numere") și o altă cale pentru a scrie în fișierul nou creat ("Numere 2"). Apoi citește datele din "Numere" și le scrie în "Numere 2". Apoi închide ambele căi.

MOVE va funcționa atît cu numere de cale (ca de ex. #4), cît și cu specificatoare de canale (ca de ex. "d";1;"Numere"). Căile standard #1, #2 și #3 nu pot fi însă specificate cu numele consacrate K, S și P.

Puteți face o copie de siguranță a fișierului "Numere" pe alt disc folosind:

```
10 MOVE "d";1;"Numere" TO "d";2;"Numere 2"
```

Instrucțiunea MOVE poate fi folosită și pentru a trimite fișiere către imprimantă. Dacă aveți o imprimantă legată la interfața serială, introduceți:

```
10 FORMAT "t",viteza
20 OPEN #4,"t"
30 MOVE "d";1;"Numere" TO #4
```

Programul Printer Server

Programul permite unui HC cuplat la o rețea să controleze o imprimantă serială. Imprimanta poate fi folosită de toate calculatoarele cuplate la rețea. Acest program este util dacă, de exemplu, un grup de utilizatori de HC posedă o singură imprimantă serială pe care vor să o împartă. Se arată totodata o utilizare mai deosebită pentru instrucțiunea MOVE.

Calculatorul folosit ca Printer Server trebuie să fie întotdeauna stația 64, și trebuie întotdeauna să facă legătura cu stația 62 (care este o stație specială de stabilire de contact). Astfel, stația emițătoare folosește temporar stația 62, și trimite numărul său real de stație, de la care va muta apoi un fișier spre canalul t. Pentru a stabili un program Printer Server introduceți:

```
10 FORMAT "n";64
20 OPEN #4;"n";62: INPUT #4;a$: CLOSE #4
30 MOVE "n";CODE a$ TO "t"
40 OPEN #4;"b": PRINT #4;CHR$ 12: CLOSE #4: RUN
```

(Linia 40 trimite un avans de pagină.)

Programul de mai jos este cel folosit de emițător. Mai întîi, emițătorul stabilește temporar stația 62. Apoi se emite numărul real de stație al emițătorului. Apoi stația emițătorului revine la numărul sau real. În final, linia 60 trimite datele care trebuiesc imprimate (în acest caz listingul).

```
10 LET statia=numar
```

unde numar este numărul de stație local.

```
20 FORMAT "n";62
30 OPEN #4;"n";64: PRINT #4;CHR$ statia: CLOSE #4
40 FORMAT "n";statia
50 OPEN #4;"n";64
60 LIST #4
70 CLOSE #4
```

CAPITOLUL 9. INSTRUCŢIUNILE CLEAR # ŞI CLS

Se recomandă folosirea instrucţiunilor CLS # şi CLEAR # în prima linie a oricărui program care foloseşte interfaţa 1.

Instrucţiunea CLEAR

Aşa cum instrucţiunea CLEAR şterge toate variabilele definite (operaţie care se execută în mod automat şi la RUN), instrucţiunea CLEAR # şterge toate canalele şi căile definite prin program, efectuînd următoarele operaţii:

- decuplează toate căile de la canalele deschise de către utilizator
- eliberează spaţiul de memorie ocupat de aceste canale. (Zona CHANS va conţine numai canalele predefinite "k", "s" şi "p".)
- cuplează căile #0, #1, #2 şi #3 la canalele standard.
- trece toate discurile în starea R/W (vezi mai jos).

Nu trebuie să se confunde efectul instrucţiunii CLEAR # cu efectul închiderii prin CLOSE # a tuturor căilor. Spre deosebire de CLOSE #, instrucţiunea CLEAR # abandonează pur şi simplu datele care se găsesc în canale. Dacă, spre exemplu, se şterge prin CLEAR # un canal de disc prin care s-a scris într-un fişier, datele din ultimul buffer vor fi pierdute, şi mai grav, nici datele care au fost deja scrise pe disc nu vor fi accesibile pentru citire.

Mesajul de eroare "Disk 'R/O'" şi CLEAR

Pentru a proteja datele înscrise pe floppy, interfaţa 1 utilizează o metodă de a preveni erorile datorate schimbărilor incorecte de floppy.

Dacă interfaţa detectează o schimbare de suport într-unul din minidrive-urile cu care a lucrat de la ultimul NEW sau CLEAR #, ea trece în mod automat discul respectiv în modul 'R/O' (numai citire).

Dacă după o astfel de schimbare se încearcă o operaţie care necesită scrierea de date pe acel disc, se va obţine mesajul de eroare "Disk 'R/O'".

Pentru a corecta aceasta situaţie, trebuie executată instrucţiunea CLEAR # imediat înainte de orice schimbare de suport într- unul din Minidrive-uri.

Instrucţiunea CLS

Efectele acestei instrucţiuni sînt similare cu execuţia comenzilor:

PRINT;: BORDER 7: PAPER 7: INK 0: CLS

ANEXA 1. JOCUL DE REŢEA

Pe discul de demonstraţie livrat odată cu sistemul există o copie a acestui joc. Numele de fişier este "net game". Programul este un bun exemplu pentru utilizarea reţelei. Părţi din el pot fi utile şi în programe scrise de dumneavoastră.

Jocul

Pentru a juca acest joc, cei doi parteneri trebuie să se gîndească fiecare la un număr între 1 şi 100. Cîştigătorul jocului este cel care ghiceşte primul numărul adversarului. La fiecare tentativă, calculatorul vă va spune cît de aproape sînteţi.

Programul

Subrutina de la linia 500 decide cine este utilizatorul 1 şi cine este utilizatorul 2. Asta este necesar pentru ca atunci cînd se transmit tentativele unul dintre voi foloseşte subrutina de la linia 1100, iar celălalt subrutina de la linia 1200, şi astfel utilizatorul 1 trimite primul, iar utilizatorul 2 primeşte primul.

Programul decide cine este utilizatorul 1 trimiţînd către celălalt calculator mesajul "1", şi apoi intrînd în ascultare pe reţea. Dacă primeşte un "1", asta înseamnă că celălalt HC a pornit programul mai tîrziu. Primul HC trimite de aceea un "2" către calculatorul adversarului, şi se face singur utilizatorul 1. Dacă, pe de alta parte, programul primeşte înapoi un "2", asta înseamnă că celălalt program era deja pornit şi în aşteptare atunci cînd programul local a trimis "1". Programul local se face singur utilizatorul 2.

Dacă cele două programe pornesc în acelaşi timp, cele doua mesaje "1" se vor ciocni pe reţea, programele se vor bloca amîndouă în aşteptare, şi este necesară întreruperea unuia dintre programe cu BREAK şi restartarea.

Programul principal schimbă numele utilizatorilor, citeşte numărul secret (care nu este trimis adversarului) şi apoi compară tentativele. Mai întîi se transmite tentativa şi apoi se afişează răspunsul.

Liniile de la 190 încolo detectează o victorie, o afişează corespunzător şi apoi oferă un alt joc.

```
10 GO SUB 500
20 PRINT;: BORDER 1: PAPER 1: INK 7: CLS
30 PRINT " Joc de ghicit numere""Introduceti mai intii un
   numar secret, apoi ghiciti-l pe al adversarului"
40 INPUT "Cum va numiti?";a$
50 PRINT ""Salut ";a$
60 GO SUB 1000 + 100*user
70 PRINT "Jucati cu ";b$
75 PRINT 'a$,b$
80 INPUT "Ghinditi-va la un numar (1 la 100)";a
90 IF a OR a100 OR aINT a THEN GO TO 80
130 INPUT "Ce numar incercati?";b
140 LET a$=STR$ b: GO SUB 1000 + 100*user
150 LET c=ABS (a-VAL b$)
160 IF c=0 THEN LET a$="Asta este" : GO TO 170
161 IF c THEN LET a$="Arde": GO TO 170
```

```
162 IF c THEN LET a$ = "Fierbinte": GO TO 170
163 IF c  THEN LET a$ = "Foarte cald": GO TO 170
164 IF c( THEN LET a$ = "Cald": GO TO 170
165 IF c« THEN LET a$ = "Rece": GO TO 170
166 LET a$ = "Gheata"
170 GO SUB 1000*100
180 PRINT b$,a$
190 IF c = 0 OR b$ = "Asta este" THEN GO TO 210
200 GO TO 130
210 IF b$ = "Asta este" THEN PRINT FLASH 1;"Victorie":
  FOR n = 0 TO 7: BORDER n: BEEP .1,n: BEEP .1,n + 16:NEXT n:
  GO TO 230
220 PRINT "Infringere": FOR n = 7 TO 0 STEP -1: BORDER n:
  BEEP .2,n: NEXT n
230 BORDER 1: INPUT "Alt joc? (d/n)";a$
240 IF a$ = "d" THEN RUN 20
250 STOP
500 OPEN #4;"n";0
510 PRINT #4;"1"
520 CLOSE #4
530 OPEN #4;"n";0
540 INPUT #4;a$
545 CLOSE #4
550 IF a$ = "1" THEN OPEN #4;"n";0: PAUSE 5: PRINT #4;"2":
  LET user = 1
560 if a$ = "2" THEN LET user = 2
570 CLOSE #4
580 FORMAT "n";user: RETURN
1100 OPEN #4;"n";3-user
1110 PRINT #4;a$
1120 CLOSE #4
1130 OPEN #4;"n";3-user
1140 INPUT #4;b$
1150 CLOSE #4
1160 RETURN
1200 OPEN #4;"n";3-user
1210 INPUT #4;b$
1220 CLOSE #4
1230 OPEN #4;"n";3-user
1240 PRINT #4;a$
1250 CLOSE #4
1260 RETURN
```


ANEXA 2. VARIABILE DE SISTEM

Pe lîngă variabilele de sistem tabelate în manualul tehnic pentru HC91, interfaţa 1 utilizează următoarele variabile:

Tip	Adresa	Nume	Conţinut
X1	23734	FLAGS3	Biţi de control interfaţa 1
X2	23735	VECTOR	Adresa folosită pentru a extinde interpretorul BASIC
X10	23737	SBRT	Rutina de paginare a ROM-urilor
2	23747	BAUD	Număr pe 16 biţi care determină rata de transfer pe linia serială calculată astfel: BAUD = (3500000/(26*baud rate))-2. O puteţi folosi pentru a stabili viteze nestandard de comunicaţie serială.
1	23749	NTSTAT	Numărul staţiei locale pe. reţea
1	23750	IOBORD	Biţii 2..0 conţin culoarea marginii ecranului în timpul I/E prin IF1. Puteţi pune orice culoare doriţi cu instrucţiunea POKE.
N2	23751	SER_FL	Spaţiu de lucru de 2 octeţi pentru interfaţa serială
N2	23753	SECTOR	2 octeţi nefolosiţi
N2	23755	CHADDT	Salvare indicator caracter curent
1	23757	NTRESP	Locaţie folosită pentru răspuns în reţea
1	23758	NTDEST	Început bloc de control în reţea. Conţine numărul staţiei destinaţie 0-64.
1	23759	NTSRCE	Numărul staţiei sursă
X2	23760	NTNUMB	Numărul blocului 0-65535
N1	23762	NTTYPE	Tip bloc: 0 - normal, 1 - ultimul (EOF)
X1	23763	NTLEN	Lungime bloc de date 0-255
N1	23764	NTDCS	Suma de control pentru blocul de date
N1	23765	NTHCS	Suma de control pentru blocul de control
N2	23766	D_STR1	Începutul primului specificator de 8 octeţi
N1	23768	?	Număr cale 0-15
N1	23769	L_STR1	Tip dispozitiv "D", "N", "T" sau "B"
N2	23770	N_STR!	Lungime nume fişier
N2	23772	F_STR!	Adresa nume fişier
N8	23774	D_STR2	Al doilea specificator de 8 octeţi folosit de MOVE şi LOAD
N1	23782	HD_00	Început zona de lucru pentru SAVE, LOAD, VERIFY şi MERGE: cod tip de date: 0 = prog, 1 = numere, 2 = şir, 3 = cod
N2	23783	HD_0B	Lungime bloc de date 0-65535
N2	23785	HD_0D	Adresa în memorie a blocului 0-65535
N2	23787	HD_0F	Lungime program fără variabile
N2	23789	HD_11	Numărul liniei de autostart
1	23791	COPIES	1 octet nefolosit
	23792		Începutul zonei CHANS
	23813		Începutul programului BASIC cu IF! dar fără canale utilizator

OBSERVAŢII

1. Inserarea variabilelor de sistem se efectuează în mod automat la prima apariţie a unei erori, a unei comenzi specifice interfeţei 1 sau în cazul mesajului OK. Această inserare poate genera mesajul Out of memory dacă cei 58 de octeţi necesari nu sînt disponibili.

2. Deschiderea unei căi sau a unui canal de disc sau reţea necesită o anumită cantitate de memorie. Un canal de disc are 306 octeţi, iar un canal de reţea are 276. Aceste canale vor fi create fie prin OPEN # sau prin MOVE. Dacă RAMTOP este prea jos, aceste comenzi pot genera mesajul de eroare Out of memory.

3. Un alt efect al introducerii variabilelor de sistem sau al creerii canalelor este mutarea programelor în cod maşină aflate în instrucţiuni REM. Puneţi întotdeauna aceste programe dupa RAMTOP.

ANEXA 3

CANALUL DE DISC

La fiecare deschidere a unui fişier prin una din instrucţiunile OPEN # sau MOVE, în zona denumită CHANS în manualul de BASIC se crează o zonă de memorie denumită canal. De obicei un canal este adresat în limbaj maşina de registrul IX. Canalul are o lungime de 306 octeţi şi conţine un bufer de 256 de octeţi.

Conţinutul canalului este următorul:

0		Adresa 8
2		Adresa 8
4		'D' sau 'D' + 80H pentru un canal ad-hoc
5		Adresa rutinei de ieşire din ROM-ul din IF1
7		Adresa rutinei de intrare din ROM-ul din IF1
9		Lungime canal, adică 306
11	CHFLAG	0 = citire, 1 = scriere posibilă din/în acest canal
12	CHDRIVE	Număr drive folosit de canal: 0 = curent, 1 = 1, 2 = 2
13	CHNAME	Numele fişierului completat cu spaţii pînă la 11 caractere. octet 9 bit 7 = r/o, octet 10 bit 7 = sys
24		20 octeţi folosiţi de sistemul de gestiune caractere
47	CHRR2	Indicator depăşire capacitate fişier în acces aleator
48	CHBYTE	Indicator caracter curent în buferul de date
50	CHDATA	256 octeţi pentru bufer

Deschiderea unui canal de disc nu crează o hartă de ocupare în memoria BASIC. Hărţile de ocupare disc există în permanenţă în memoria RAM instalată pe IF1 memorie comutată împreună cu ROM-ul din IF1.

CANALUL DE REŢEA

La deschiderea unei căi către reţea se creează o zonă de memorie denumită canal în spaţiul indicat de variabila de sistem CHANS. Această zonă este adresată în limbaj maşină de registrul IX. Canalul are o lungime de 276 octeţi şi conţine un buffer de 255 de octeţi.

Conţinutul canalului este următorul:

0		Adresa 8
2		Adresa 8
4		'N' pentru OPEN # sau 'N' + 80H pentru MOVE
5		Adresa rutinei de ieşire din ROM-ul din IF1
7		Adresa rutinei de intrare din ROM-ul din IF1

9		Lungime canal adică 276
11	NCIRIS	Numărul stației partenere în comunicație
12	NCSELF	Numărul stației locale la deschiderea canalului
13	NCNUMB	Numărul blocului 0-65535
14	NCTYPE	Tipul pachetului de date: 0 = normal, 1 = ultimul
15	NCOBL	Numărul de octeți în blocul de date
17	NCDCS	Suma de control pentru blocul de date.
18	NCHCS	Suma de control pentru blocul de control
19	NCCUR	Deplasament în buffer pentru ultimul caracter transferat
20	NCIBL	Numărul de octeți utili în buffer
21	NCB	255 octeți pentru buferul de date

CANALUL DE SERIALĂ

La deschiderea unei căi către interfața serială se crează o zonă de memorie denumită canal în spațiul indicat de variabila de sistem CHANS. Această zonă este adresată în limbaj mașină de registrul IX. Canalul are o lungime minimă de 11 octeți.

Conținutul canalului este următorul:

0	Adresa 8
2	Adresa 8
4	'B' sau 'T'
5	Adresa rutinei de ieșire din ROM-ul din IF1
7	Adresa rutinei de intrare din ROM-ul din IF1
9	Lungime canal adică 11

ANEXA 4. CONEXIUNI INTERFAȚA SERIALĂ

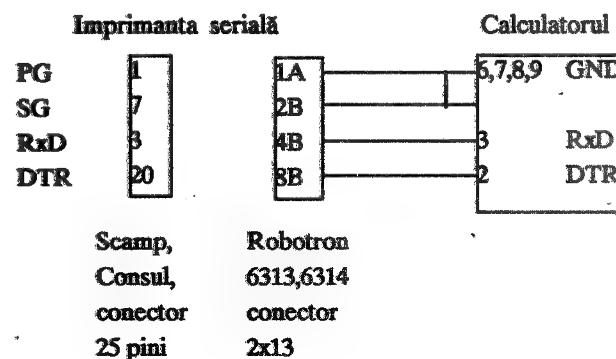
Conectorul de interfață serială este folosit precum urmează:

1. CTS ieșire, este la nivel ridicat dacă este 'gata'
2. DTR intrare, trebuie să fie la nivel ridicat pentru 'gata'
3. RxData (ieșire)
4. TxData (intrare)
5.
6. Masa
7. Masa
8. Masa
9. Masa

Pentru conectare cu o interfață standard CCITT V24, în capătul celălalt al cablului trebuie folosit un conector cu 25 de pini cablat în felul următor:

1. Masa (PG)
2. TxData
3. RxData
5. CTS
6. +12v (DSR)
7. Masa(SG)
20. DTR

Pentru cuplarea imprimantei sînt necesare semnalele RxD, DTR și bineînțeles Masa. Cuplarea se face fie cu un cablu ecranat cu 2 fire "calde", fie cu trei fire simple introduse, eventual, într-un varniș. De reținut că interfața lucrează în protocol DTR. Pentru aceasta se fac conexiunile:



ANEXA 5. MESAJE DE EROARE

Instrucțiunile implementate de interfața 1 generează mesaje de eroare diferite de mesajele de eroare generate de ROM-ul din placa de bază. Aceste mesaje vor fi urmate de numărul liniei și numărul comenzii din linia care a generat eroarea.
Aceste noi mesaje de eroare sînt listate mai jos în ordine alfabetică:

CODE error

Ați încercat să încărcați (LOAD) un bloc de cod a cărui lungime este mai mare decît lungimea specificată de instrucțiunea LOAD.

Disk error

În timpul execuției unei operații de intrare/ieșire pe disc a apărut o eroare care nu a putut fi reparată prin reîncercări.

Disk full

Ați încercat să scrieți date într-un disc care nu avea suficient spațiu liber. Reîncercați programul cu un alt disc, sau eliberați spațiu pe discul curent stergînd fișierele de care nu mai aveți nevoie.

Disk 'R/O'

Ați încercat să efectuați o operație de scriere pe un suport schimbat, fără să comunicați calculatorului prin CLEAR # faptul că ați terminat de lucrat cu vechiul suport. Introduceți CLEAR # și apoi repetați comanda.

Disk 'write' protected

Ați încercat o operație de scriere pe un disc care are montată protecția la scriere. Îndepărtați protecția și apoi reîncercați.

File not found

Ați încercat o operație asupra unui fișier inexistent, sau ați încercat o operație CAT pe un disc fără nici un fișier.

File 'R/O'

Ați încercat să ștergeți sau să scrieți un fișier care are atributul de protejat la scriere (octetul 9 din nume bitul 7=1). Deprotejați fișierul dacă sînteți sigur că vreți să-l modificați.

Invalid device expression

Ați specificat un dispozitiv diferit de k, s, p, d, n, b sau t. Același mesaj se obține dacă ați folosit punct- virgulă în loc de virgulă pentru unul din specificatorii k, s sau p.

Invalid drive number

Ați specificat un număr de Minidrive mai mare ca 2, sau ați specificat numărul 0 (Minidrive-ul curent), înainte de a-l declara printr-un apel explicit.

Invalid name

Numele fișierului este fie un șir vid, fie are mai mult de unsprezece caractere.

Invalid station number

S-a specificat un număr de stație în afara domeniului 0-64 (1-64 pentru instrucțiunea FORMAT).

Invalid stream number

Numărul de cale specificat este în afara domeniului 0-15.

MERGE error

Ați încercat să comasați date sau cod. MERGE funcționează numai cu programe.

Missing baud rate

Lipsește rata de transfer în instrucțiunea FORMAT "b" sau "t".

Missing drive number

Lipsește numărul minidrive-ului.

Missing name

Lipsește numele fișierului.

Missing station number

Lipsește numărul stației în rețea.

Program finished

Ați încercat să executați o linie dincolo de ultima linie din program. Acest mesaj de eroare va apare dacă executați un GO TO urmat de un număr de linie mai mare decît ultima linie din program. Va apare de asemenea dacă introduceți RUN fără a avea un program în memorie.

Reading a 'write' file

Încercați să citiți date dintr-un fișier disc inexistent, sau dintr-un canal care a fost deja folosit pentru scriere.

Stream already open

Ați încercat să deschideți o cale care a mai fost folosită pentru un canal de tip nou (d, n, t sau b). Calea poate fi deschisă numai după ce a fost închisă.

Verification has failed

Există diferențe între fișierul salvat și programul, datele sau codul existente în memorie.

Writing to a 'read' file

Ați încercat să scrieți într-un fișier disc existent. Fișierul existent trebuie mai întîi șters, dacă nu este nevoie de el. Altfel trebuie utilizat un fișier nou.

Wrong file type

Ați încercat să încărcați (LOAD) un fișier de date sau cod ca pe un program sau invers un program ca pe un fișier de date sau cod.

ANEXA 6. BASIC-UL EXTINS

Interfața 1 extinde BASIC-ul existent deja în HC. Extensiile și adăugirile sînt rezumate mai jos.

Căile

Căile sînt specificate prin #n unde n este un număr în domeniul 0-15. Căile 0, 1, 2 și 3 sînt de obicei folosite de BASIC. Caracterul # este parte din cuvîntul cheie pentru instrucțiunile OPEN # și CLOSE #.

Canalele

Există șapte tipuri de canale în BASIC-ul extins: claviatura (k), ecranul (s), imprimanta (p), interfața serială pentru texte (t), interfața serială binară (b), rețeaua (n) și discul (d).

Fiecare canal este specificat prin litera lui care poate fi majusculă sau nu. Rețeaua și discul au nevoie de informații suplimentare pentru a specifica complet canalul.

Un canal de rețea necesită un număr de stație, așa încît un specificator de rețea are forma "n";x unde x este numărul stației în domeniul 0-64.

Un canal de disc necesita un număr de minidrive și un nume de fișier care trebuie să fie un șir cu 1 pînă la 11 caractere.

Minidrive curent

Primul minidrive la care se face acces după NEW sau CLEAR # va deveni ceea ce se numește minidrive-ul curent. Din acest moment înainte acest minidrive poate fi specificat și cu numărul 0.

Dacă se încearcă folosirea specificatorului 0 înainte de a defini minidrive-ul curent, se va semnala eroarea Invalid drive number.

Caracterul ? în nume de fișier

Este bine să nu folosiți caracterul ? în numele de fișiere, pentru că acest caracter are alt rol. El este folosit de sistemul de fișiere de post de "Jolly Jocker" (wild card), putînd să înlocuiască orice alt caracter, dar unul și numai unul.

Dacă de exemplu trebuie să ștergeți fișierele cu numele "nume0", "nume1", "nume2" și nu aveți alte fișiere cu nume de forma "numex" (unde x este orice caracter), puteți folosi o singură comandă de forma ERASE "d",1,"nume?" care șterge toate cele trei fișiere.

Dacă într-un OPEN # specificați parțial un nume de fișier existent (folosind caracterul "?") va fi folosit primul fișier al cărui nume se potrivește.

Instrucţiuni

CAT y	Listează toate numele fişierelor aflate pe discheta din minidrive-ul y. Lista este prezentată în ordinea din catalogul discului şi este urmat de dimensiunea spaţiului liber pe dischetă în kilocteţi. ATENŢIE! Dacă discheta a fost schimbată şi nu s-a introdus NEW sau CLEAR #, numărul reprezentînd spaţiul liber nu reflectă realitatea.
CAT #z;y	Trimite catalogul floppy-ului din minidrive-ul y către calea z.
CAT y;"cc...c"	Listează numele fişierelor care se potrivesc cu şirul de caractere "cc...c", care poate conţine "?" pentru specificări ambigue.
CAT #z;"cc...c"	Ca mai sus, dar trimite lista către calea z.
CLEAR #	Readuce sistemul de căi şi canale la starea de după NEW - există numai canalele standard k, s, p şi sînt deschise numai căile standard #0, #1, #2 şi #3. Eventualele date existente în canalele utilizator sînt ignorate, spaţiul de memorie fiind eliberat fără remuşcări.
CLOSE #cale	Desface legătura dintre calea specificată şi orice canal. Dacă există date blocate prin scriere în bufferul canalului, atunci acestea sînt fie transmise (pe reţea) sau înregistrate (pe dischetă).
CLS #	Readuce ecranul 10n starea de după NEW. Border alb, PAPER alb, INK negru, ecran şters.
ERASE "d";y;"nume"	Şterge fişierele specificate de nume aflate pe discul din minidrive-ul y. Numele poate conţine caracterul "?" pentru specificaţii ambigue.
FORMAT "d";y	Pregăteşte un floppy din minidrive-ul y pentru a fi utilizat în BASIC
FORMAT "n";x	Stabileşte numărul staţiei pe reţea la x.
FORMAT "t";x FORMAT "b";x	Stabileşte viteza de comunicaţie pentru interfaţa serială la x (x trebuie ales dintre vitezele standard de comunicaţie 50, 110, 300, 600, 1200, 2400, 4800, 9600, 19200).
INKEY#cale	Întoarce un singur caracter sub forma unui şir dacă cel puţin unul este disponibil sau întoarce şirul vid "" dacă nu există caracter disponibil din calea respectivă.Această instrucţiune are sens doar dacă calea este legată la un canal de reţea sau de interfaţă serială.
INPUT#cale;var	Citeşte variabila var din calea specificată. Calea trebuie să fi fost deschisă înainte către un canal de intrare. Este important să reţineţi că orice lement de PRINT care apare în instrucţiunea INPUT va fi scris către această cale. Aceasta este de obicei necesar atunci cînd se citesc date de la claviatură. Reţineţi de asemenea că separatorul "," scrie un caracter. Opţiunea LINE este disponibilă ca mai înainte.
LOAD *canal opţiuni	Încarcă programul, datele sau codul de la canalul specificat. Se pot folosi numai canalele "b", "n" sau "d". Toate opţiunile existente pentru LOAD sînt disponibile şi pentru LOAD *
MERGE *canal opţiuni	La fel ca LOAD, doar că nu şterge liniile de program sau variabilele decît pentru a face loc pentru unele noi, cu acelaşi număr de linie sau nume.
MOVE sursa TO destinaţie	Mută datele de la sursă către destinaţie. Sursa şi destinaţia pot fi numere de cale sau canale. Comanda se termină numai la întîlnirea unui indicator de sfîrşit de fişier în sursă: aceasta se poate întîmpla doar dacă sursa este un canal de reţea sau disc, sau altfel o cale legată la un astfel de canal. Dacă sursa sau destinaţia sînt specificate drept canale, atunci acestea sînt deschise la început şi închise la terminarea transferului.
OPEN #cale,canal	Leagă calea specificată la canalul specificat pentru a permite programului BASIC să citească sau să scrie din/în acel canal. Calea trebuie să fie închisăsau deschisă către unul din canalele k, s sau p.

PRINT #cale...	Tipăreşte secvenţa de print către calea specificată. Calea trebuie să fi fost deschisă în prealabil către un canal de ieşire. Secvenţa de print poate avea aceeaşi sintaxă ca mai înainte şi poate conţine şi alte elemente de tipul #cale.
SAVE *canal opţiuni	Salvează programul, datele sau codul către canalul specificat. Pot fi folosite numai canalele "b", "n" sau "d". Toate opţiunile existente la SAVE sînt disponibile şi la SAVE *
VERIFY *canal opţiuni	La fel ca LOAD cu excepţia faptului că datele nu sînt încărcate în memorie, ci doar comparate cu ceea ce există deja acolo.

ANEXA 7. Lansare CP/M pe HC 91

Pentru a lansa sistemul de operare CP/M pe calculatorul HC91 se introduce în drive-ul 1 un disc cu sistemul de operare CP/M, se apasă butonul de RESET şi apoi se execută linia BASIC:

PRINT USR 14446

Pe ecran apare mesajul:

56K CP/M ver 2.2/10
A>

Din acest moment de la claviatură se poate introduce o comandă CP/M (Vezi manualul de operare CP/M).

Claviatura sau Console Input

Claviatura are 40 de taste: 26 de litere, 10 cifre, 2 shift, SPACE şi CR. Cu toate acestea de la claviatură se pot genera 127 din cele 128 de coduri ASCII - singurul care lipseşte este NULL = 00H care nu este necesar în utilizarea programelor sub CP/M. Cele 127 de coduri se obţin fie prin taste simple fie prin combinaţii de două sau trei taste. Pentru anumite coduri, care se obţin printr-o combinaţie standard de trei taste, mai există şi o 'scurtătură' de numai două taste. Este cazul codurilor de control des folosite de WordStar şi alte programe pentru poziţionare cursor.

În tabela următoare sînt prezentate tastele sau combinaţiile de taste care trebuiesc apăsate pentru a obţine cele 127 de coduri. Toate codurile sînt în hexa. Coloana Nume ASCII conţine modul de referire din manuale la acest cod. Codificarea ^C de exemplu este echivalentă cu ceea ce în anumite manuale se denumeşte prin combinaţia CONTROL/C sau CTRL/C. Coloana Standard conţine tasta sau combinaţia de taste standard pentru culegerea codului. CS este prescurtarea pentru CapsShift iar SS este prescurtarea pentru SymbolShift. Pentru a culege ^C se apasă mai întîi cu cîte un deget de la mîna stîngă şi respectiv dreapta cele două taste SHIFT şi apoi ţinînd în continuare tastele SHIFT apăsate se acţionează clapa C. Similar se culeg şi celelalte coduri care necesită combinaţii de taste.

Tastatura este prevazută cu o facilitate de autorepetare. O tasta sau o combinaţie de taste care este menţinută mai multă vreme apăsată va declanşa după un timp o secvenţă de coduri identice cu codul caracterului apăsat.

Dacă trebuiesc introduse multe litere mari în secvenţă, se poate acţiona combinaţia de taste 'CS 2' - adică CapsShift şi 2 - pentru a obţine un mod de funcţionare al claviaturii în care toate literele generează codurile corespunzatoare literelor mari indiferent dacă tasta CapsShift este apăsată sau nu. În acest mod cursorul afişat pe ecran are dimensiunea egală cu înălţimea unei litere mari. Pentru a reveni la modul normal de lucru, se mai acţionează odată combinaţia 'CS 2'. Cursorul se micşorează şi se pot introduce din nou litere mici.

În tabela următoare sînt prezentate cele 127 de coduri care se pot obţine de la claviatură, în varianta standard şi în varianta scurtă, dacă aceasta există.

Cod hexa	Nume ASCII	STANDARD	Scurtătura	Comentarii
01	^A	CS SS A	CS 4	
02	^B	CS SS B		
03	^C	CS SS C	CS 3	
04	^D	CS SS D	CS 8	Săgeata la dreapta
05	^E	CS SS E	CS 7	Săgeata în sus
06	^F	CS SS F	CS 9	
07	^G	CS SS G		BELL
08	^H	CS SS H		BACKSPACE
09	^I	CS SS I		TAB
0A	^J	CS SS J		LINE FEED
0B	^K	CS SS K		
0C	^L	CS SS L		FORM FEED
0D	^M	CS SS M		CARRIAGE RETURN
0E	^N	CS SS N		
0F	^O	CS SS O		
10	^P	CS SS P		
11	^Q	CS SS Q		
12	^R	CS SS R		
13	^S	CS SS S	CS 5	Săgeata la stînga
14	^T	CS SS T		
15	^U	CS SS U		
16	^V	CS SS V		
17	^W	CS SS W		
18	^X	CS SS X	CS 6	Săgata în jos
19	^Y	CS SS Y		
1A	^Z	CS SS Z		
1B	^[	CS 1		ESCAPE
1C	^\	SS E		
1D	^]	SS I		
1E	^	SS Q		
1F	^	SS W		
20	SP	SPACE		
21	!	SS 1		
22	"	SS P		
23	#	SS 3		
24	$	SS 4		
25	%	SS 5		

26	&	SS 6		
27	'	SS 7		
28	(	SS 8		
29	)	SS9		
2A	*	SS B		
2B	+	SS K		
2C	,	SS N		
2D	-	SS J		
2E	.	SS M		
2F	/	SS V		
30 - 39	0 - 9	0 - 9		
3A	:	SS Z		
3B	;	SS O		
3C	-	SS R		
3D	=	SS L		
3E		SS T		
3F	?	SS C		
40	@	SS 2		
41 - 5A	A - Z	CS A - CS Z	în CAPS LOCK mode	
5B	[	SS Y		
5C	\	SS D		
5D	]	SS U		
5E	^	SS H		
5F		SS 0		
60	'	SS X		
61 - 7A	a - z	a - z		
7B	{	SS F		
7C	|	SS S		
7D	}	SS G		
7E	~	SS A		
7F	DEL	CS)		

Ecranul sau Console Output

Caracterele trimise spre rutina de Console Output sînt interpretate şi au ca efect fie afişarea unui caracter pe ecran - dacă sînt tipăribile - fie au un efect de control al ecranului - dacă sînt caractere sau secvenţe de control recunoscute - fie sînt ignorate sau au un efect nedeterminat - dacă sînt caractere de control sau secvenţe nerecunoscute.

Toate caracterele trimise spre Console Output sînt mascate cu masca 7F, adică se ignoră bitul 7.

Caracterele tipăribile sînt cuprinse între 20 hexa (BLANK) şi 7E (tilda ~). Caracterul 7F afişează în mod excepţional un bloc de puncte.

Caracterele de control sînt cuprinse între codurile 00 şi 1F. Nu toate sînt valide. Cele invalide sînt ignorate. Codurile de control valide sînt prezentate în continuare.

Caracter	Mnemonică	Descriere
07	BELL	Scurt sunet pe difuzorul calculatorului
08	BACKSPACE	Mută cursorul la stînga, fără să afecteze caracterele de pe ecran
09	TAB	Mută cursorul la dreapta pînă într-o coloană multiplu de 8
0A	LINE FEED	Mută cursorul cu o linie în jos. Efectuează scroll dacă este pe ultima linie
0B	CURSOR UP	Mută cursorul cu o linie în sus
0C	CURSOR RIGHT	Mută cursorul la dreapta, fără a afecta informaţia de pe ecran
0D	CR	Mută cursorul în prima coloană a liniei curente
18	CLEAR SCREEN	Şterge ecran şi aduce cursor în colţul stînga sus
1A	CLEAR SCREEN	Şterge ecran şi aduce cursor în colţul stînga sus
1B	ESCAPE	primul caracter al unei secvenţe de control, denumită şi secvenţa de ESCape. Secvenţele de ESCape vor fi prezentate mai jos.
1E	HOME	Aduce cursor în colţul stînga sus

În continuare sînt prezentate secvenţele de ESCape recunoscute. O secvenţă de ESCape invalidă, va avea un efect nedeterminat asupra ecranului. Din acest motiv, sînteţi îndemnaţi să nu folosiţi secvenţe de ESCape invalide.

Secvenţa	Mnemonica	Descriere
1B 13	DELTE LINE	Şterge linia în care se află cursorul. Liniile aflate pe ecran mai jos vor efectua scroll în sus, ocupînd locul liniilor de deasupra lor. În ultima linie de pe ecran va apare o linie goală. Cursorul nu-şi modifică poziţia
1B 1A	INSERT LINE	Linia curentă şi liniile următoare se mută cu o linie mai jos, ultima linie de pe ecran dipărînd. Linia curentă va fi goală. Cursorul nu-şi modifică poziţia.
1B 41 <linie=20..37> <coloana=20..5F>	SET CURSOR	Cursorul va sări în linia şi în coloana specificată de caracterele 3 şi respectiv 4 ale secvenţei. Caracterul care specifică linia ia valori între 20 şi 37 hexa, ceea ce corespunde liniilor 0 (sus) şi 17 (jos). Caracterul care specifică coloana ia valori între 20 şi 5F hexa, ceea ce corespunde coloanelor 0 (stînga) şi 3F (dreapta)

1B 42	CLEAR END LINE	Toate caracterele aflate între poziţia curentă a cursorului inclusiv şi ultimul de pe linia curentă vor fi înlocuite cu BLANK. Cursorul nu-şi modifică poziţia.
1B 49 <culoare=30..37>	SET INK	Schimbă culoarea cernelii cu care vor fi scrise în continuare caracterele pe ecran. Nu are nici un efect asupra caracterelor care sînt deja pe ecran.
1B 50 <culoare=30..37>	SET PAPER	Schimbă culoarea hîrtiei (fondului) cu care vor fi scrise în continuare caracterele pe ecran. Nu are nici un efect asupra caracterelor care sînt deja pe ecran.
1B 51 <culoare=30..37>	SET BORDER	Schimbă culoarea marginii ecranului
1B 52 <culoare=30..37 sau 20..27>	SET I/O BRDER	Schimbă culoarea pe care o ia marginea ecranului în timpul operaţiilor de I/O. Dacă codul de culoare este în intervalul 30..37, schimbarea culorii va fi însoţită de un mic click pe difuzor. Dacă codul de culoare este în intervalul 20..27, schimbarea marginii se va face în linişte.
1B 59	CLEAR END SCREEN	Toate caracterele aflate între poziţia curentă a cursorului inclusiv şi ultimul din ecran vor fi înlocuite cu BLANK. Cursorul nu-şi modifică poziţia
1B 6A	SET INVERSE	După această secvenţă toate caracterele vor fi afişate în 'invers video'
1B 6B	CLEAR INVERSE	După această secvenţă toate caracterele vor fi afişate în 'normal video'
1B 70	SEMIG ON	Activare set alternat de caractere. Setul alternat de caractere va fi afişat mai jos
1B 71	SEMIG OFF	Dezactivare set alternat de caractere, adică revenire la setul normal
1B 72	CLICK ON	Fiecare citire de tastă de la claviatură va fi însoţită de un click la difuzor după această secvenţă
1B 73	CLICK OFF	După această secvenţă, citirea caracterelor de la claviatură se va face în linişte
1B 74	CURSOR ON	Cursorul este vizibil pe ecran
1B 75	CURSOR OFF	Cursorul este invizibil pe ecran

După comutarea în setul alternat, caracterele tipăribile şi anume cele cuprinse în intervalul 20..2E hexa, vor afişa alte forme decît în setul standard, şi anume un set special conceput pentru încadrarea tabelelor. Aceste caractere sînt denumite în literatură caractere semigrafice. Codurile de la 2B la 7F vor afişa forme nedeterminate.

Codurile pentru caracterele semigrafice sînt:

Caracter	Mnemonica
20	bara verticală
21	-I
22	colţ drepta sus
23	colţ stînga jos
24	_I_
25	T
26	I-
27	-
28	+

Pentru a reveni în setul normal de caractere se trimite spre Console Output secvenţa **1B 71**.

Cum se pregăteşte o dischetă pentru lucrul în CP/M

Aşa cum s-a arătat, la alimentare sau după RESET, Hc este în BASIC. Pentru trecerea în CP/M introducem discheta de CP/M în unitatea de disc şi dăm comanda:

PRINT USR 14446 «CR»

Apare, în stînga sus, mesajul:

56k CP/M ver. 2.2/10
A»

În acest moment sîntem în CP/M. Pentru a vedea ce conţine discheta, se dă comanda:

A»dir «CR»

Apare menu-ul cu ce conţine discheta.
Dorim să pregătim o dischetă în vederea imprimării de programe. În UDF avem discheta sursă. Scriem:

A»format a: «CR»

Apare mesajul:

Insert new disk in drive A:
Strike ENTER when ready

Schimbaţi discheta sursă cu o dischetă nouă şi apăsaţi «CR». Începe formatarea. După ce se termină, apare mesajul:

Format complete
Format another (Y/N)?

Scriem:

n
A»

Încărcăm SYSGEN de pe discul sursă şi îl punem pe discul nou, proaspăt formatat. În UDF avem discheta sursă. Scriem:

A »sysgen

Apare:

SYSGEN VER 2.0
SOURCE DRIVE NAME (OR RETURN TO SKIP)

Scriem :

a

şi apare mesajul:

SOURCE ON A, THEN TYPE RETURN

Apăsăm « CR » şi apare:

FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT)

Apăsăm:

a

şi apare mesajul:

DESTINATION ON A, THEN TYPE RETURN

Schimbăm discheta sursă cu discheta nouă ş apăsăm « CR ». Apare:

FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT)

Apăsăm « CR » şi ieşim din SYSGEN.

Dăm comanda :

A »dir « CR »

Pe ecran apare:

NO FILE

Apăsăm RESET apoi, după iniţializarea în BASIC apăsăm:

PRINT USR 14446 « CR »

Apare:

56k CP/M VER. 2.2/10
A »

Scriem:

A »dir « CR »

Apare:

NO FILE

Discheta este acum gata oentru a scrie pe ea programe, fişiere, date sau cod.